

Analysis of Track Fusion Using the Reduced State Estimator

George J. Foster

Strategic and Weapon Control Systems Department
Naval Surface Warfare Center Dahlgren Division
Dahlgren, VA, U.S.A.
george.foster@ieee.org

Abstract – *We evaluate the performance of track fusion of two-state tracks based on individual track estimates. The individual track estimates are evaluated using a Reduced State Estimator that provides a separate estimate of the filtered random error and the bias resulting from target acceleration. By separating the bias and random portions of the errors, the filters and fusion process can be optimized to minimize cost functions specific to each application.*

Keywords: Tracking, track fusion, Kalman filtering, estimation, reduced state estimator, cost function, optimization.

1 Introduction

The two-state Kalman filter has found wide acceptance within the target tracking community [1, 2, 3]. However, the default optimization of the Kalman filter to minimizing the mean squared error of the diagonal of the error covariance matrix does not match the needs of all systems. Many weapon systems and air traffic control systems require a filter response to target accelerations that minimizes the combination of the filter noise output and the lag in response to acceleration. In fact, many system designers used simplified forms of the Kalman filter such as the alpha-beta filter to allow alternate gain selection techniques that would optimize the filter gains according to different criteria [4, 5, 6]. Some techniques of translating the alternate gain selection criteria into modified process noise values have allowed the use of Kalman filters to achieve the desired cost function optimization as long as the tracking rates and measurement noises were kept within nominal values [7].

Most recently, techniques have been developed to modify the Kalman filter structure to account for the bias that develops as a result of tracking an accelerating target with a two-state filter designed for nominally constant-velocity targets [8]. This reduced state estimator does not estimate the acceleration value itself as done using input estimation or in a three-state filter [1]. Instead, it calculates the theoretical lag of the filter in response to acceleration and then modifies the gain selection process in the covariance equations to achieve a specific balance of the optimization between minimizing lag and

minimizing noise. This bound on maximum bias due to acceleration provided by the reduced state estimator does not suffer from the delays that are seen when using other augmentation techniques in [1] such as input estimation.

In this paper, we evaluate the performance of the reduced order estimator when it is used as the filter for the individual tracks in a track fusion problem. Since the reduced state estimator is relatively unknown in the community, we first restate the development of this filter before developing the fusion technique. We then demonstrate the performance by running simple simulations that illustrate the features of the reduced state estimator and fusion. Finally, we provide a summary of the analysis and ideas for future development.

2 Reduced State Estimator

The basic concept of the reduced state estimator is to divide the covariance estimate which traditionally is estimated by the Kalman filter into two parts: the random noise portion of the estimate and the bias term based on assumed target acceleration. So while the Kalman filter requires only that the state vector x and covariance matrix P be passed from one estimation cycle to the next; the reduced state estimator relies on three objects: the state vector x , the random noise covariance matrix M , and the filter time response matrix D . We explore the equations of the reduced state estimator next.

2.1 Update Equations

In a similar manner to the Kalman filter, the reduced order estimator algorithm has three parts:

- a. Predict state to measurement time
- b. Compute gains
- c. Update state

We define the state vector to be

$$x = \begin{bmatrix} x \\ y \\ z \\ \vdots \\ \vdots \\ \vdots \end{bmatrix} \quad (1)$$

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE JUL 2011		2. REPORT TYPE		3. DATES COVERED 00-00-2011 to 00-00-2011	
4. TITLE AND SUBTITLE Analysis of Track Fusion Using the Reduced State Estimator				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Surface Warfare Center Dahlgren Division, Strategic and Weapon Control Systems Department, Dahlgren, VA				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES Presented at the 14th International Conference on Information Fusion held in Chicago, IL on 5-8 July 2011. Sponsored in part by Office of Naval Research and U.S. Army Research Laboratory.					
14. ABSTRACT We evaluate the performance of track fusion of two-state tracks based on individual track estimates. The individual track estimates are evaluated using a Reduced State Estimator that provides a separate estimate of the filtered random error and the bias resulting from target acceleration. By separating the bias and random portions of the errors, the filters and fusion process can be optimized to minimize cost functions specific to each application.					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 7	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

The update from step k to $k+1$ is then as typical for two-state filters

$$x_p = \phi x_k \quad (2)$$

where

$$\phi = \begin{bmatrix} 1 & 0 & 0 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 & \tau & 0 \\ 0 & 0 & 1 & 0 & 0 & \tau \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

and

$$\tau = t_{k+1} - t_k \quad (4)$$

The random error covariance matrix is updated in the same fashion as the covariance matrix from the Kalman filter

$$M_p = \phi M_k \phi^T \quad (5)$$

The innovative part of the reduced state estimator is the handling of the bias from acceleration lag. The final part of the prediction step of the estimator is to propagate the time response matrix, which estimates the filter lag, forward to the predicted time by

$$D_p = \phi D_k + G\lambda \quad (6)$$

where the state error matrix, G , defines how the non-random acceleration introduces a potential error to the predicted state vector as

$$G = \begin{bmatrix} \frac{\tau^2}{2} & 0 & 0 \\ 0 & \frac{\tau^2}{2} & 0 \\ 0 & 0 & \frac{\tau^2}{2} \\ \tau & 0 & 0 \\ 0 & \tau & 0 \\ 0 & 0 & \tau \end{bmatrix} \quad (7)$$

and λ is the acceleration design factor, usually the maximum expected acceleration from the target.

To compute the filter gains, we first combine the random and bias portions of our error estimates into a total error matrix

$$S = n^2 M_p + D_p D_p^T \quad (8)$$

where n is selected to provide the desired level of confidence in the estimate, where $n=2$ or $n=3$ are typical of critical applications and higher values of n may be used in safety critical systems. The gains are then computed using a slightly modified version of the Kalman gain equation

$$K = SH^T (HSH^T + n^2 R)^{-1} \quad (9)$$

where R is the measurement noise matrix, and the measurement transformation matrix is given as

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (10)$$

Finally, to perform the update state step of the process, we first update the state itself using the measurement z

$$x_{k+1} = x_p + K(z - Hx_p) \quad (11)$$

The random error covariance matrix is updated exactly as expected from the Kalman filter

$$M_{k+1} = (I - KH)M_p(I - KH)^T + KRK^T \quad (12)$$

and the filter time response matrix is updated using

$$D_{k+1} = (I - KH)D_p \quad (13)$$

2.2 Initialization

The reduced state estimator can be initialized in a similar manner as other constant-velocity filters. If we initialize using the first two measurements, then we can set the state estimate to be

$$x_2 = \begin{pmatrix} z_2 \\ \frac{z_2 - z_1}{\tau} \end{pmatrix} \quad (14)$$

and the random error covariance matrix to be

$$M_2 = \begin{pmatrix} R_2 & \frac{R_2}{\tau} \\ \frac{R_2}{\tau} & \frac{R_1 + R_2}{2\tau^2} \end{pmatrix} \quad (15)$$

Finally, the time response matrix is initialized to

$$D_2 = \begin{bmatrix} \frac{\tau^2}{2} & 0 & 0 \\ 0 & \frac{\tau^2}{2} & 0 \\ 0 & 0 & \frac{\tau^2}{2} \\ \tau & 0 & 0 \\ 0 & \tau & 0 \\ 0 & 0 & \tau \end{bmatrix} \lambda \quad (16)$$

2.3 Discussion

While the derivation of the reduced state estimator is explained in depth in [8], it is easy to see that the reduced state estimator does not deviate far from the form of the Kalman filter, but the deviations are significant in its ability to handle accelerations in a manner consistent to weapon and tracking system design. A typical fire control tracking engineer is given the baseline performance of the sensor and the maximum maneuverability of the targets under track in addition to the knowledge of how the filtering error transforms to weapon system performance. A major difference in reality of fire control tracking from the design assumptions in the Kalman filter is the behavior

Kalman Filter	Reduced State Estimator
Predict state	
$x_p = \phi x_k$ $P_p = \phi P_k \phi^T$	$x_p = \phi x_k$ $M_p = \phi M_k \phi^T$ $D_p = \phi D_k + G \lambda$
Compute gains	
$S = P_p$ $K = SH^T (HSH^T + n^2 R)^{-1}$	$S = n^2 M_p + D_p D_p^T$ $K = SH^T (HSH^T + n^2 R)^{-1}$
Update state	
$x_{k+1} = x_p + K(z - Hx_p)$ $P_{k+1} = (I - KH)P_p(I - KH)^T + KRK^T$	$x_{k+1} = x_p + K(z - Hx_p)$ $M_{k+1} = (I - KH)M_p(I - KH)^T + KRK^T$ $D_{k+1} = (I - KH)D_p$

Figure 1. Comparison of the Kalman filter to the reduced state estimator

of target acceleration. Typical targets do not accelerate in a Gaussian random fashion. Instead, they accelerate in bursts in one direction for a period of time, and then return to the nominal profile. By accounting for the maneuvering lag separately, the reduced state estimator keeps the estimate of the random error covariance lower. A comparison of the Kalman filter and reduced state estimator is shown in Figure 1.

3 Reduced State Fusion

To design the fusion algorithm for the reduced state estimator, we must first decide the type of fusion we

desire. Figure 2 illustrates the two general techniques available to us. While measurement fusion allows for the most use of the information in each track update, this technique is more susceptible to errors in sensor alignment and noise outliers in a single sensor measurement can corrupt the fused estimate for a considerable period of time.

Instead, we choose track fusion, which allows us to develop a separate state estimate for each sensor individually, each with its own filter. At each filter update, we can then update the overall state estimate by fusing the individual sensor based state estimates using the same

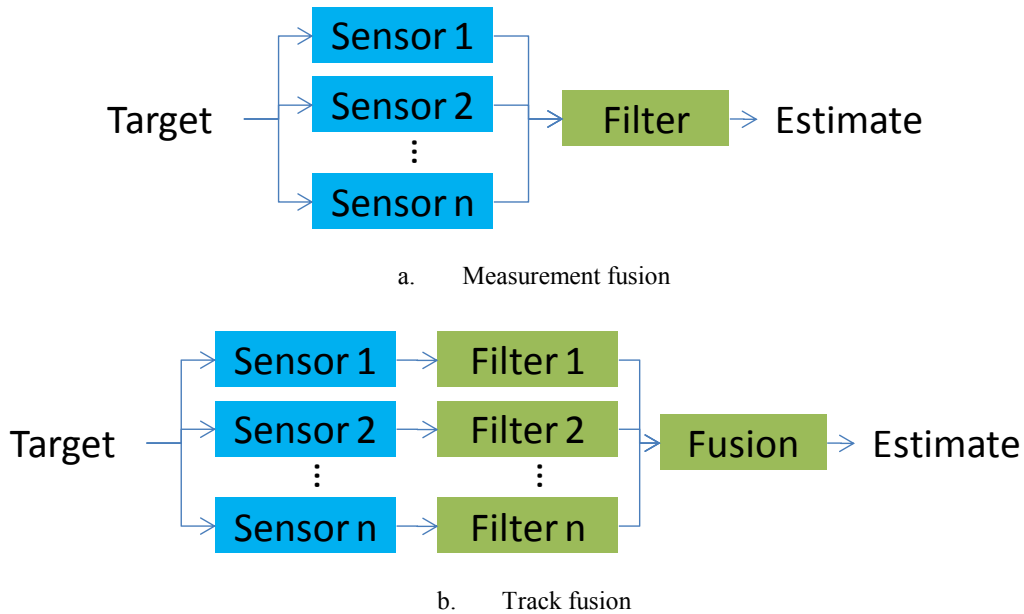


Figure 2. Block diagrams of measurement fusion vs. track fusion

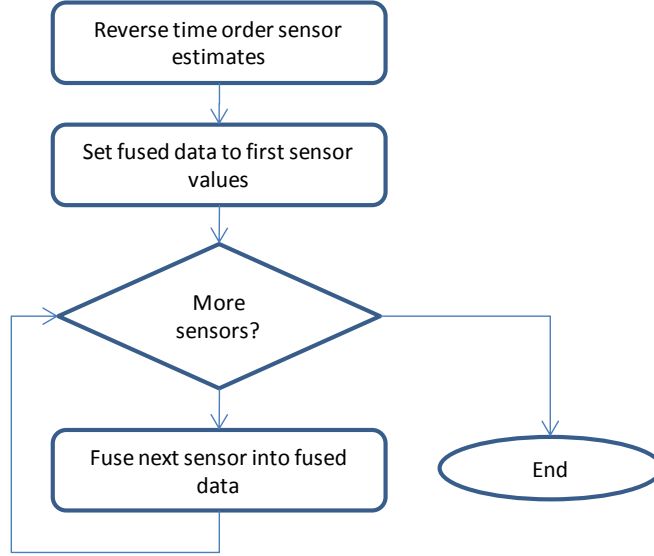


Figure 3. Block diagram of recursive fusion process

basic form as the reduced state estimator. We use the basic two-track fusion algorithm found in [9] or [10], but with modifications to account for the bias estimate.

3.1 Fusion algorithm

To fuse the outputs of two reduced state estimators, we must first designate one as the reference estimate. We select the estimate with the most recent time and designate the state estimate, time, random error covariance, and time response matrix as x_f , t_f , M_f , and D_f , the initial fused values. We then select our sensor with the next most recent time and error matrices x_n , t_n , M_n , and D_n .

We first predict the second sensor data to the fused time t_f by using the same basic equations given previously in (2) through (7)

$$x_p = \phi x_n \quad (17)$$

$$M_p = \phi M_n \phi^T \quad (18)$$

$$D_p = \phi D_n + G \lambda \quad (19)$$

where this time the values of ϕ and G are calculated based on the update duration of

$$\tau = t_f - t_n \quad (20)$$

We construct the same approximation to our cost function to calculate the fusion gain as in (8) and (9)

$$S = n^2 M_f + D_f \quad (21)$$

$$R = n^2 M_p + D_p \quad (22)$$

and finally calculate the gain using

$$K = S(S + R)^{-1} \quad (23)$$

Finally, to update the states and error matrices, we update the fused state estimate as

$$x_f = x_f + K(x_p - x_f) \quad (24)$$

$$M_f = (I - KH)M_f(I - KH)^T + KM_pK^T \quad (25)$$

$$D_f = (I - KH)D_f + KD_p \quad (26)$$

The process repeats for each sensor in reverse time sequence if there are more than two sensors. In that case, the output of the fusion results from fusing each sensor is used as our reference estimate for when fusing the next sensor. Figure 3 contains a block diagram of this process.

3.2 Discussion

The track fusion algorithm presented here is the simplest method of fusion that honors the cost function approach. The astute reader can easily notice that the filter time response matrices have a high degree of correlation from one sensor to another. Further work is required to properly decouple the contributions to prevent overestimating the amount of acceleration lag is present in the fused estimate.

4 Sample Performance

4.1 One Dimensional Example

To easily grasp the impact of performing track fusion using reduced state estimators, we consider a simple one dimensional example using a constant velocity target. In this case, we have two sensors tracking the target with measurement reports occurring at the same exact instant. For each of our cases, we perform 1000 Monte Carlo runs to base our evaluations of the mean and standard deviations of our estimates. In Figure 4 we see the comparison of the standard deviation of the position estimate for the single target estimate to the fused estimate. We see the similar comparison for the velocity

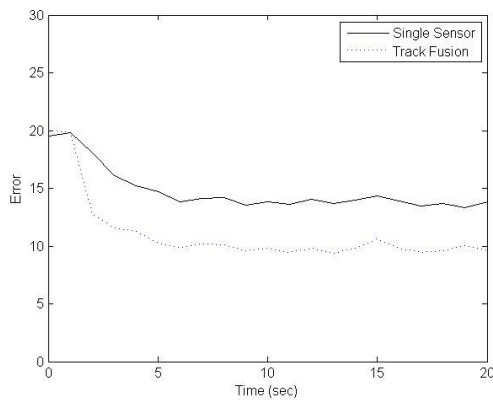


Figure 4. Standard deviation of position estimates

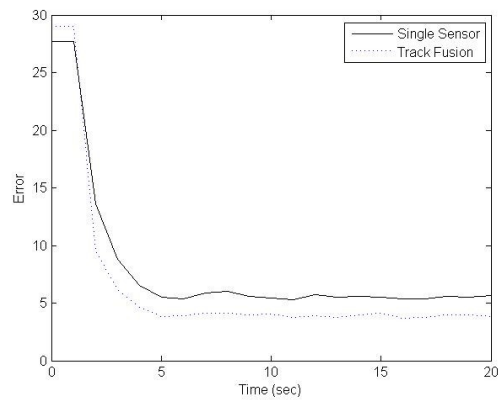


Figure 5. Standard deviation of velocity estimates

estimate in Figure 5. In both cases, we see the expected noise reduction from the addition of the second sensor of $1/\sqrt{2}$.

We then modify the scenario to introduce a target with a constant acceleration. In Figures 6 and 7, we again see a comparison of the noise performance of the single sensor estimate to the fused estimate. We are now interested in the acceleration performance of the fused estimate. Figures 8 and 9 show the mean performance of the filters for position and velocity. As expected, the fused estimate shows no improvement over the single source estimate, since all the sensor updates are occurring at the same time and responding to the same acceleration input. Figures 10 and 11 compare the estimate of the filter lag in the D matrix to the empirical lag from the simulation. For both position and velocity, we see good agreement between the estimate of the maximum lag and the actual filter lag.

4.2 Typical Maneuvering Targets

Now that we have established the agreement between the estimates of the errors and lags with the simulation results, we can move our example into a more realistic realm. We consider two scenarios: In the first scenario, the target is flying directly at the first sensor, with the second sensor positioned approximately equidistant from the target on a vector perpendicular to the line from the target to the first sensor. The target flies a constant velocity leg, then accelerates toward the first sensor on a constant acceleration leg, and then continues to fly on toward the first sensor at the new velocity. In the second scenario, the sensor platforms are laid out identically to the first. In this scenario, after completing the first constant velocity leg, the target maneuvers in a constant speed turn toward the second sensor before returning to a constant velocity leg heading directly toward the second sensor.

As expected, Figures 12 and 13 demonstrate the minor improvement gained in the velocity estimate in the x direction using the fusion technique.

5 Summary

The reduced state estimator shows promise as a means for balancing the requirements of filter lag versus noise performance. We have demonstrated that the approach of separating the lag and noise terms within the filter can be extended to the track fusion process as well. The simple simulation performed herein demonstrated the basic capability of the reduced state estimator to provide an estimate of the filter lag in addition to the random covariance that is typical of Kalman filters. Considerable work needs to be made on fully realizing the track fusion algorithms that can take advantage of the high correlation of the bias (lag) terms from one sensor's track to the next in a manner similar to [11].

Acknowledgements

John Becker of Lockheed Martin in New Jersey has provided years of fruitful collaboration in the area of fire control tracking. The author would also like to acknowledge the recent contributions of Gregg Bock of Lockheed Martin in expanding the basic theory of the reduced state estimator to full featured fire control tracking.

References

- [1] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, Academic Press, 1988.
- [2] Y. Bar-Shalom and X. Li, *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, 1993.
- [3] S. S. Blackman, *Multi-Target Tracking with Radar Applications*, Artech House, 1986.
- [4] S. L. Burton, G. J. Foster, and J. E. Gray, "A Method for Characterizing Track Goodness when Design

Mismatch Occurs,” Twenty-Ninth Southeastern Symposium on System Theory, Cookeville, TN, March 9-11, 1997.

[5] J. E. Gray and G. J. Foster, “Filter coefficient selection using design criteria,” Twenty-Eighth Southeastern Symposium on System Theory, Baton Rouge, LA, March 31 - April 2, 1996.

[6] J. E. Gray and G. J. Foster, “An Alternative Method to Using Plant Noise as a Means of Selecting Filter Gains in a Complex Tracking Environment,” IEEE 2000 International Radar Conference, Washington, DC, May 7-12, 2000.

[7] J. E. Gray and G. J. Foster, “An Extension of the Tracking index concept to non-Kalman filter selection techniques,” Thirtieth Southeastern Symposium on System Theory, Morgantown, WV, March 8-10, 1998.

[8] P. Mookerjee and F. Reifler, “Reduced State Estimator for Systems with Parametric Inputs,” IEEE Transactions on Aerospace and Electronic Systems, Vol. 40, No. 2, pp. 446-461, April 2004.

[9] R. A. Singer and A. J. Kanyuck, “Computer control of multiple site track correlation,” Automatica, Vol. 7, pp. 455-463, 1971.

[10] J. A. Roecker and C. D. McGillem, “Comparison of two-sensor tracking methods based on state vector fusion and measurement fusion,” IEEE Transactions on Aerospace and Electronic Systems, Vol. 24, No. 4, pp. 447-449, 1988.

[11] Y. Bar-Shalom and L. Campo, “The effect of the common process noise on the two-sensor fused-track covariance,” IEEE Transactions on Aerospace and Electronic Systems, Vol. 22, No. 6, pp. 803-804, 1986.

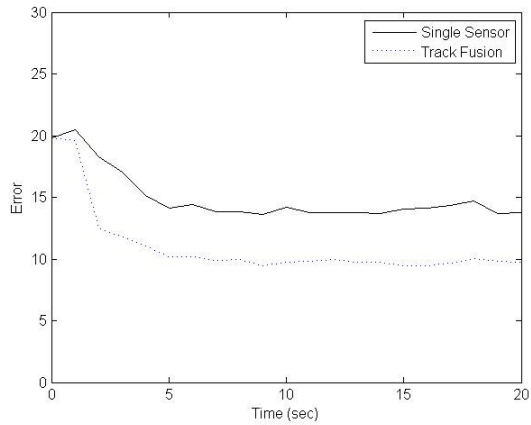


Figure 6. Standard deviation of position estimates for CA target

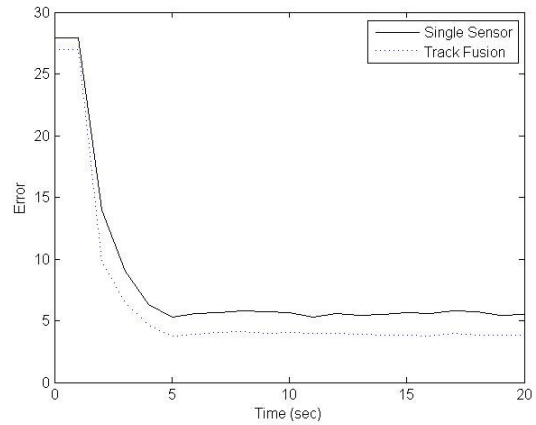


Figure 7. Standard deviation of velocity estimates for CA target

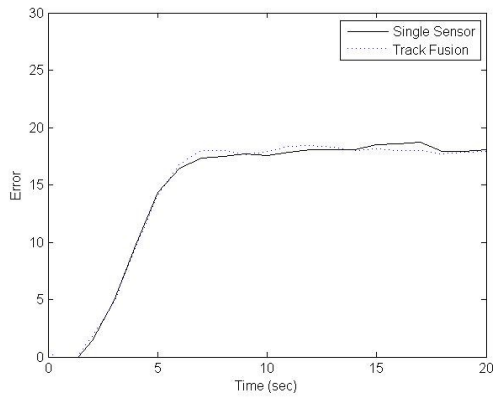


Figure 8. Mean error (lag) of position estimates for CA target

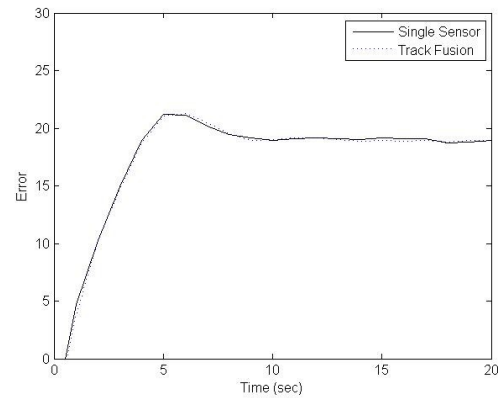


Figure 9. Mean error (lag) of velocity estimates for CA target

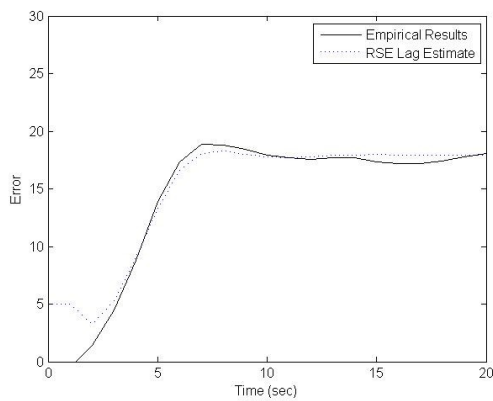


Figure 10. Comparison of estimated position lag to mean error for CA target

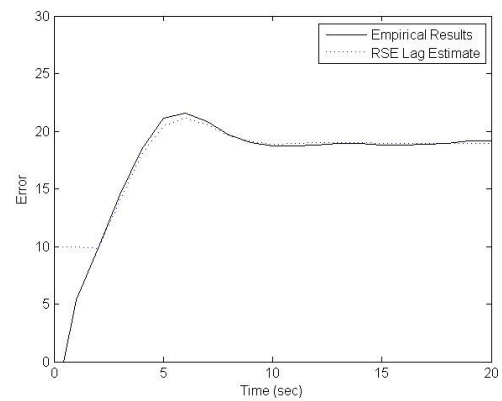


Figure 11. Comparison of estimated velocity lag to mean error for CA target